Stability Bonus Regularization for Model Selection Under Positive-Class Distribution Shift

David C. Liu
Independent Researcher
50685071@proton.me

Abstract

Standard cross-validation selects the hyperparameter configuration that maximizes held-out performance on the *training* distribution. When the positive class in training is a biased subset of the true positive population—a common scenario in medical screening, hiring, and credit scoring—the selected model can overfit to the observed cluster and fail on unseen positive subgroups at deployment. We propose **Stability Bonus (SB) regularization**, a lightweight modification to cross-validated model selection that rewards configurations whose training and validation scores are close, thereby favoring wider decision boundaries that are less sensitive to the specific positive-class cluster seen during training. In controlled experiments on six datasets (three synthetic with calibrated distribution shift, three real-world with simulated bias), SB with τ=0.10 improves test AUC by +6.9% ($p < 0.0001$, d=3.48) and +3.1% (p=0.027, d=1.25) on the two hardest synthetic benchmarks, and improves unseen-subgroup AUC by up to +7.1% on synthetic benchmarks. Results on real datasets are mixed, with improvements on some datasets and degradation on others. Critically, SB does *not* help when signal is strong or when training positives already match the deployment distribution. Class weighting, the standard remedy for imbalance, is the *worst* performer under distribution shift because it amplifies the biased positives. We release all code and data for reproducibility.

Keywords: model selection, distribution shift, cross-validation, regularization, positive-class bias

1 Introduction

Consider a medical diagnostic model trained on hospital records where all confirmed positive cases came from a single urban clinic. The model learns to recognize the symptom profile of *that clinic's patient population*—the "circle"—but at deployment it must identify positives drawn from the broader population—the "oval." This is not a class-imbalance problem in the traditional sense: the issue is not that positives are rare, but that the *observed* positives are a non-representative subset of the *true* positive class.

This form of positive-class distribution shift arises in many domains:

- **Hiring:** Historical hires (positives) over-represent certain demographic groups, so models trained on them fail to identify qualified candidates from underrepresented groups [17].
- **Medical diagnosis:** Training data drawn from one hospital or one demographic group may not represent the full disease phenotype [19].
- **Credit scoring:** Approved loans (positives) reflect past lending criteria that excluded certain populations [11].

Standard remedies for class imbalance—SMOTE [3], ADASYN [12], class weighting [7]—address the *frequency* of the minority class but not its *representativeness*. Indeed, class weighting can make the problem *worse*: upweighting biased positives causes the model to fit them even harder.

We propose a different intervention point: *model selection*. Rather than modifying the training data or loss function, we modify the criterion by which hyperparameter configurations are ranked during

cross-validation. Our method, **Stability Bonus (SB) regularization**, adds a small bonus to the validation score of configurations whose training–validation gap is below a threshold τ. The intuition is that models which overfit to the biased positive cluster will have high training AUC but lower validation AUC (because validation folds, while drawn from the same biased distribution, exhibit fold-to-fold variance that penalizes tight cluster fitting). SB rewards configurations where this gap is small, which empirically correlates with wider decision boundaries that generalize to unseen positive subgroups.

Our contributions:

1. We formalize SB regularization and provide geometric intuition for when it should help (Section 3).

2. We conduct rigorous experiments with held-out test sets, Bonferroni-corrected significance tests, and effect sizes across six datasets (Section 4).

3. We show that SB significantly improves test AUC under weak-signal, high-dimensional distribution shift, while honestly reporting where it does *not* help (Section 5).

4. We demonstrate that SB consistently selects more regularized hyperparameter configurations, confirming the "wider boundary" hypothesis (Section 5).

2 Related Work

2.1 Class Imbalance

Data-level methods modify the training distribution via oversampling [3, 10, 12], undersampling [24, 26, 15], or hybrid approaches [16]. Algorithm-level methods adjust loss functions via cost-sensitive learning [7] or threshold calibration. Comprehensive surveys are provided by Fernández et al. [8] and Krawczyk [14]. These methods address *class frequency* imbalance but assume the observed minority class is representative of the true minority class—an assumption we relax.

2.2 Distribution Shift and Domain Adaptation

Covariate shift methods reweight training instances so that the training distribution matches the test distribution [21, 23]. Domain adaptation methods learn representations invariant across domains [9]. Quiñonero-Candela et al. [19] provide a taxonomy of dataset shift types. Our work addresses a narrower problem—shift within the positive class only—and operates at the simpler level of model selection rather than feature learning.

2.3 Cross-Validation and Model Selection

Cross-validation remains the standard for model selection [22, 1, 13], though it can produce overoptimistic estimates under class imbalance [20, 27]. Varma and Simon [25] and Cawley and Talbot [2] analyze selection bias when the same data is used for both model selection and evaluation. SB regularization can be viewed as a modified selection criterion within nested cross-validation that incorporates stability information.

2.4 Fairness in Machine Learning

The practical motivation for our work overlaps with algorithmic fairness [6, 11, 17]: when positive training data is demographically biased, models systematically underperform on underrepresented groups. Our approach does not explicitly encode fairness constraints but may complement fairness-aware methods by selecting models that generalize beyond the observed positive cluster.

3 Method

3.1 Standard Cross-Validated Model Selection

In standard k-fold cross-validation, each hyperparameter configuration θ is evaluated by training on $k-1$ folds and computing a validation score $S_v^{(i)}$ on fold i. The configuration is ranked by its mean validation score:

$$\text{Score}_{\text{std}}(\theta) = \frac{1}{k} \sum_{i=1}^{k} S_v^{(i)}(\theta) \tag{1}$$

The configuration with the highest score is selected.

3.2 Stability Bonus Regularization

We augment the selection criterion by rewarding configurations whose training–validation gap is small. Let $S_t^{(i)}$ and $S_v^{(i)}$ be the training and validation scores for fold i, and let $g^{(i)} = |S_t^{(i)} - S_v^{(i)}|$ be the generalization gap. The Stability Bonus score for fold i is:

$$\mathrm{SB}^{(i)}(\theta) = \begin{cases} S_v^{(i)} \cdot (1 + b^{(i)}) & \text{if } g^{(i)} < \tau \\ S_v^{(i)} & \text{otherwise} \end{cases} \quad (2)$$

where the bonus magnitude is:

$$b^{(i)} = \frac{\tau - g^{(i)}}{\tau} \cdot \beta \quad (3)$$

The threshold τ controls the maximum gap eligible for a bonus, and β controls the maximum bonus magnitude. The final ranking score is $\mathrm{Score}_{\mathrm{SB}}(\theta) = \frac{1}{k}\sum_{i=1}^{k} \mathrm{SB}^{(i)}(\theta)$.

Key properties:

- **Non-punitive:** SB never reduces a score below the raw validation score. It can only promote stable configurations, never demote them.
- **Bounded:** The maximum bonus is β (e.g., 20%), preventing extreme distortion of the ranking.
- **Monotonic in stability:** Smaller gaps receive larger bonuses, providing a smooth gradient toward stable configurations.

3.3 Geometric Intuition: Circle Inside Oval

Suppose the true positive class occupies an ellipsoidal region (the "oval") in feature space, but training positives are drawn only from a spherical subregion (the "circle"). A model that fits tightly around the circle will achieve high training AUC but will miss positive instances in the oval's tails. During cross-validation, each fold's positive subsample will be a slightly different slice of the circle, so a tight-fitting model will show fold-to-fold variance in its training–validation gap. A model with a wider decision boundary—one that encompasses more of the oval—will be less sensitive to which slice of the circle appears in each fold, producing a smaller and more stable gap. SB exploits this signal by preferring configurations with small gaps.

3.4 Comparison: Gap Penalty

An alternative formulation penalizes all train–validation gaps:

$$S_{\mathrm{gap}} = S_v - \alpha|S_t - S_v| \quad (4)$$

This can inadvertently penalize high-performing configurations that happen to have legitimately high training scores. SB avoids this failure mode by only *rewarding* stability rather than *punishing* instability.

4 Experimental Design

4.1 Datasets

We use six datasets: three synthetic with controlled distribution shift and three real-world with simulated positive-class bias.

4.1.1 Synthetic Datasets

Generated using `sklearn.make_classification` [18] with varying difficulty:

- **synth_med_50d:** 50 features (10 informative), shift = 0.5, bias at 35th percentile. Moderate difficulty; serves as the "easy" control.
- **synth_weak_80d:** 80 features (8 informative), shift = 0.4, bias at 30th percentile. High dimensionality with weak signal.
- **synth_vweak_100d:** 100 features (6 informative), shift = 0.3, bias at 25th percentile. The hardest setting: very weak signal, high noise dimensionality, strong bias.

In all synthetic datasets, training positives are drawn from a biased subregion (below the bias percentile on a latent shift dimension), while test positives are drawn from the full population.

4.1.2 Real-World Datasets

We simulate positive-class bias on three OpenML datasets:

- **Adult (income):** Positive = income >$50K. Bias: training positives are males only; test includes all genders.
- **Credit-g:** Positive = bad credit risk. Bias: training positives are age < 30 only; test includes all ages.
- **Diabetes:** Positive = diabetes diagnosis. Bias: training positives are low-BMI only; test includes all BMI levels.

4.2 Methods Compared

Six model selection strategies:

1. **Standard CV:** Rank by mean validation AUC (baseline).
2. **SB (τ=0.05):** Stability Bonus with τ=0.05, β=0.2.
3. **SB (τ=0.10):** Stability Bonus with τ=0.10, β=0.2.
4. **SB (τ=0.20):** Stability Bonus with τ=0.20, β=0.2.
5. **Class-Weighted:** Standard CV with `scale_pos_weight` set inversely proportional to class frequency.
6. **Gap Penalty:** $S_v - |S_t - S_v|$ with α=1.

4.3 Protocol

All experiments use XGBoost [4] as the base learner. We search over 60 hyperparameter configurations randomly sampled from the grid spanning: `max_depth` $\in \{3, 5, 7, 10\}$, `min_child_weight` $\in \{1, 3, 5, 10, 20\}$, `reg_alpha` $\in \{0, 0.01, 0.1, 1\}$, `reg_lambda` $\in \{0.1, 1, 5, 10\}$, `n_estimators` $\in \{100, 150, 200\}$.

Each configuration is evaluated via stratified 5-fold cross-validation. Each complete experiment is repeated 5 times with different random seeds. The primary metric is **test-set AUC** computed on a held-out test set that includes the *full* positive population (not just the biased subset). We also report **unseen-subgroup AUC** (performance on positive instances from the part of the oval outside the circle) and the **generalization gap** (train AUC − test AUC).

4.4 Statistical Testing

All pairwise comparisons are against Standard CV. We report mean differences, Cohen's d effect sizes [5], and Bonferroni-corrected p-values for 5 simultaneous comparisons per dataset.

5 Results

5.1 Test AUC

Table 1 reports mean test AUC (± standard deviation across 5 repeats) for all methods and datasets. The best result per dataset is **bolded**.

5.2 Statistical Significance

Table 2 reports Bonferroni-corrected p-values and Cohen's d for SB (τ=0.10) versus Standard CV, the comparison of primary interest.

Key finding: SB (τ=0.10) produces statistically significant improvements on the two hardest synthetic datasets (synth_vweak_100d: +6.9%, d=3.48; synth_weak_80d: +3.1%, d=1.25) but not on the easier synthetic dataset or the real-world datasets. On the adult dataset, SB (τ=0.10) is significantly *worse*, though the practical difference is small (−0.9%).

5.3 Unseen Subgroup Performance

Table 3 reports AUC on the positive instances that were excluded from training (the "oval minus the circle").

Table 4 reports significance tests for unseen-subgroup AUC.

The pattern mirrors overall test AUC: SB significantly improves unseen-subgroup performance on the hard synthetic datasets (+7.1% and +3.6%) but not on the easy synthetic or real-world datasets.

5.4 Generalization Gap

Table 5 reports the generalization gap (train AUC − test AUC). Lower is better; it indicates less over-

Table 1: Test AUC (mean ± std across 5 repeats). Best per dataset in **bold**. SB = Stability Bonus.

Dataset	Standard CV	SB (τ=0.05)	SB (τ=0.10)	SB (τ=0.20)	Class-Wt.	Gap Pen.
synth_vweak_100d	.455±.111	.524±.124	**.524±.106**	.503±.100	.435±.108	.518±.097
synth_weak_80d	.660±.070	.676±.077	**.692±.053**	.687±.049	.639±.076	.687±.049
synth_med_50d	.835±.023	.831±.019	**.837±.017**	.836±.021	.821±.027	.837±.021
diabetes	.642±.046	**.685±.038**	.652±.049	.652±.049	.540±.060	.652±.049
adult	.869±.004	.860±.003	.860±.003	**.871±.004**	.865±.004	.865±.007
credit-g	**.674±.025**	.651±.026	.620±.058	.624±.059	.673±.026	.620±.058

Table 2: SB (τ=0.10) vs. Standard CV. Significant results ($p < 0.05$) in **bold**.

Dataset	ΔAUC	d	p (Bonf.)
synth_vweak_100d	**+.069**	**3.48**	**<.0001**
synth_weak_80d	**+.031**	**1.25**	**.027**
synth_med_50d	+.002	0.19	1.000
diabetes	+.010	0.52	1.000
adult	−.009	−3.22	<.0001
credit-g	−.054	−1.13	.059

fitting.

SB reduces the generalization gap in five of six datasets (credit-g is essentially tied). Even on datasets where SB does not improve test AUC (adult, credit-g), it selects models with smaller train–test discrepancies. This confirms that SB is functioning as designed—selecting more conservative models—even when that conservatism does not translate to higher test AUC on a particular dataset.

5.5 Class-Weighted Performance Under Shift

A striking result across all datasets is that **class weighting is the worst-performing method under distribution shift**. On synth_vweak_100d, class-weighted achieves 0.435 AUC versus 0.455 for Standard CV ($p < 0.001$, $d = -1.78$). On diabetes, class-weighted drops to 0.540 versus 0.642 for Standard CV ($p < 0.0001$, $d = -2.67$).

This is expected: class weighting increases the effective weight of positive training instances, which under distribution shift means *amplifying the bias*. The model fits even more tightly to the observed positive cluster.

5.6 Hyperparameter Selection Patterns

Table 6 compares the hyperparameters selected by Standard CV versus SB (τ=0.10) on the synthetic datasets, averaged across repeats.

The most consistent pattern is in `reg_alpha` (L1 regularization): SB selects configurations with `reg_alpha` = 1.0 across all three synthetic datasets, compared to near-zero values for Standard CV. Higher L1 regularization zeroes out weak or noisy features, producing sparser models with wider effective decision boundaries—exactly the mechanism hypothesized in Section 3.3.

6 Discussion

6.1 When SB Helps—and When It Does Not

Our results reveal a clear pattern: SB helps when three conditions co-occur:

1. **Weak signal:** Few informative features relative to total features (6/100 or 8/80).
2. **High dimensionality:** Many noise features that enable overfitting.
3. **Strong distribution shift:** The training positive cluster is substantially smaller than the true positive region.

When signal is strong (synth_med_50d: 10/50 informative features) or the distribution shift is mild, Standard CV already selects models that generalize adequately, and SB provides no additional benefit.

On real-world datasets, results are mixed. Diabetes shows a promising +4.3% AUC improvement with SB (τ=0.05), but this is not statistically significant (p=0.98) due to high variance across repeats (n=5). Adult shows a small but significant *decrease*

Table 3: Unseen Subgroup AUC (mean ± std). Best per dataset in **bold**.

Dataset	Standard CV	SB (τ=0.05)	SB (τ=0.10)	SB (τ=0.20)	Class-Wt.	Gap Pen.
synth_vweak_100d	.447±.113	.518±.127	**.518±.108**	.497±.102	.427±.110	.512±.099
synth_weak_80d	.646±.073	.663±.081	**.681±.056**	.676±.051	.622±.080	.676±.051
synth_med_50d	.825±.022	.822±.018	**.830±.016**	.827±.019	.810±.026	.828±.018
diabetes	.527±.067	**.589±.061**	.540±.078	.540±.078	.384±.090	.540±.078
adult	.442±.019	.411±.024	.410±.025	**.466±.026**	.419±.024	.434±.045
credit-g	**.490±.020**	.464±.013	.408±.074	.412±.077	.487±.023	.408±.074

Table 4: Unseen Subgroup AUC: SB (τ=0.10) vs. Standard CV.

Dataset	ΔAUC	d	p (Bonf.)
synth_vweak_100d	**+.071**	**3.56**	**<.0001**
synth_weak_80d	**+.036**	**1.35**	**.013**
synth_med_50d	+.005	0.36	1.000
diabetes	+.013	0.43	1.000
adult	−.032	−1.87	<.001
credit-g	−.082	−0.96	.163

with SB (τ=0.05, 0.10), though the best SB variant (τ=0.20) recovers to match Standard CV. Credit-g shows consistent degradation.

We attribute the mixed real-world results to two factors. First, our bias simulation (restricting training positives by a single feature) may not capture the complexity of real-world distribution shift, where bias operates across multiple correlated features. Second, real datasets have stronger inherent signal than our hardest synthetic settings, placing them in the regime where SB is unnecessary.

6.2 Why Class Weighting Fails Under Distribution Shift

Class weighting addresses the question: "Are there enough positive examples?" Distribution shift poses a different question: "Are the positive examples representative?" By upweighting the biased positive instances, class weighting effectively says: "These biased examples are even more important than you thought." The model responds by fitting them more tightly, *widening* the gap between training and deployment performance. This is visible in the generalization gap (Table 5): class-weighted models have the largest train–test discrepancies on 5 of 6 datasets.

6.3 The Wider Decision Boundary Mechanism

The hyperparameter analysis (Table 6) confirms the hypothesized mechanism. SB consistently selects higher `reg_alpha`, which applies L1 regularization to the leaf weights in XGBoost. This has two effects: (1) it zeroes out splits on noisy features that happen to correlate with the biased cluster, and (2) it produces smaller leaf weights, which correspond to less confident predictions—i.e., wider, softer decision boundaries.

The result is a model that captures the broad direction of the signal without memorizing the specific location of the biased positive cluster.

6.4 Sensitivity to τ

The three τ values tested (0.05, 0.10, 0.20) show that τ=0.10 is generally the best or near-best choice. Too small (τ=0.05) and the bonus is rarely triggered, reducing SB's effect. Too large (τ=0.20) and configurations with substantial overfitting can receive bonuses. The moderate value τ=0.10 strikes a balance, and we recommend it as a default.

6.5 Limitations

- **Small number of repeats.** With only 5 repeats, confidence intervals are wide, especially on the real-world datasets. Larger-scale experiments would strengthen conclusions.

- **Simulated bias on real datasets.** Our bias simulation (restricting positives by a single demographic feature) is a coarse proxy for real-world distribution shift, which is typically more subtle and multidimensional.

Table 5: Generalization Gap (train AUC − test AUC, mean ± std). Lower is better. Best per dataset in **bold**.

Dataset	Standard CV	SB (τ=0.05)	SB (τ=0.10)	SB (τ=0.20)	Class-Wt.	Gap Pen.
synth_vweak_100d	.442±.091	.308±.137	**.302±.073**	.339±.068	.474±.089	.316±.056
synth_weak_80d	.261±.087	.223±.105	**.172±.034**	.183±.029	.274±.072	.183±.029
synth_med_50d	.114±.014	.099±.029	**.072±.010**	.099±.015	.124±.017	.094±.020
diabetes	.321±.058	**.262±.063**	.297±.052	.297±.052	.427±.084	.297±.052
adult	.101±.004	.082±.003	.082±.004	**.077±.004**	.085±.004	.080±.004
credit-g	.299±.022	**.252±.028**	.297±.070	.301±.065	.308±.023	.297±.070

Table 6: Average selected hyperparameters on synthetic datasets: Standard CV vs. SB (τ=0.10).

Dataset	Param	Std CV	SB
synth_vweak_100d	reg_alpha	0.026	**1.000**
	min_child_wt	4.4	2.2
	max_depth	3.0	3.0
synth_weak_80d	reg_alpha	0.022	**1.000**
	min_child_wt	7.2	3.0
	max_depth	3.2	3.0
synth_med_50d	reg_alpha	0.060	**1.000**
	min_child_wt	5.8	3.0
	max_depth	3.0	3.0

- **Single base learner.** We tested only XGBoost. The effectiveness of SB with other model families (linear models, neural networks) remains unknown.
- **Binary classification only.** Extension to multi-class settings is not explored.
- **Fixed β.** We fixed β=0.2 and varied only τ. Joint optimization of (τ, β) may yield better results.
- **No distribution-shift baselines.** We compare against standard model selection and class-imbalance methods but not against methods specifically designed for distribution shift, such as importance weighting or distributionally robust optimization. Future work should benchmark against these approaches.

7 Conclusion

Stability Bonus regularization is *not* a general-purpose class imbalance technique—our experiments make that clear. It does not help when signal is strong, dimensionality is moderate, or the training positive class already represents the deployment distribution.

What SB *does* provide is a simple, principled intervention for a specific and practically important failure mode: model selection under positive-class distribution shift in high-dimensional, weak-signal settings. Under these conditions, SB (τ=0.10) improves test AUC by up to 6.9% ($p < 0.0001$) and unseen-subgroup AUC by up to 7.1%, while standard class weighting—the intuitive first remedy—actually *worsens* performance.

The mechanism is transparent: SB selects more regularized models (higher L1 penalty, sparser trees) that form wider decision boundaries, avoiding overfitting to the biased positive cluster.

Future work should validate SB on real-world datasets with natural (not simulated) distribution shift, explore adaptive τ and β schedules, and investigate integration with explicit fairness constraints for high-stakes applications in hiring, lending, and medical diagnosis.

Reproducibility

All code, data generation scripts, and experiment configurations are available at `https://github.com/davidcliu/stability-bonus-regularization`.

Acknowledgments

The author thanks the open-source scientific Python ecosystem [18, 4]. Computational assistance from Claude (Anthropic) was used for code development and manuscript preparation. All experimental designs, analyses, and conclusions are the author's own.

References

[1] Sylvain Arlot and Alain Celisse. A survey of cross-validation procedures for model selection. *Statistics Surveys*, 4:40–79, 2010.

[2] Gavin C Cawley and Nicola LC Talbot. On over-fitting in model selection and subsequent selection bias in performance evaluation. *Journal of Machine Learning Research*, 11:2079–2107, 2010.

[3] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357, 2002.

[4] Tianqi Chen and Carlos Guestrin. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794, 2016.

[5] Jacob Cohen. Statistical power analysis for the behavioral sciences. 1988.

[6] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness through awareness. In *Innovations in Theoretical Computer Science*, pages 214–226, 2012.

[7] Charles Elkan. The foundations of cost-sensitive learning. *International Joint Conference on Artificial Intelligence*, 17(1):973–978, 2001.

[8] Alberto Fernández, Salvador García, Mikel Galar, Ronaldo C Prati, Bartosz Krawczyk, and Francisco Herrera. Learning from imbalanced data sets. 2018.

[9] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *International Conference on Machine Learning*, pages 1180–1189, 2015.

[10] Hui Han, Wen-Yuan Wang, and Bing-Huan Mao. Borderline-SMOTE: A new over-sampling method in imbalanced data sets learning. In *International Conference on Intelligent Computing*, pages 878–887. Springer, 2005.

[11] Moritz Hardt, Eric Price, and Nathan Srebro. Equality of opportunity in supervised learning. In *Advances in Neural Information Processing Systems*, volume 29, 2016.

[12] Haibo He, Yang Bai, Edwardo A Garcia, and Shutao Li. ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In *IEEE International Joint Conference on Neural Networks*, pages 1322–1328. IEEE, 2008.

[13] Ron Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. *International Joint Conference on Artificial Intelligence*, 14(2):1137–1145, 1995.

[14] Bartosz Krawczyk. Learning from imbalanced data: open challenges and future directions. *Progress in Artificial Intelligence*, 5(4):221–232, 2016.

[15] Miroslav Kubat and Stan Matwin. Addressing the curse of imbalanced training sets: One-sided selection. In *International Conference on Machine Learning*, volume 97, pages 179–186, 1997.

[16] Xu-Ying Liu, Jianxin Wu, and Zhi-Hua Zhou. Exploratory undersampling for class-imbalance learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, 39(2):539–550, 2009.

[17] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *ACM Computing Surveys*, 54(6):1–35, 2021.

[18] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[19] Joaquin Quiñonero-Candela, Masashi Sugiyama, Anton Schwaighofer, and Neil D Lawrence. Dataset shift in machine learning. 2009.

[20] Miriam Seoane Santos, Jastin Pompeu Soares, Pedro Henriques Abreu, Helder Araujo, and Joao Santos. Cross-validation for imbalanced datasets: Avoiding overoptimistic and overfitting approaches. *IEEE Computational Intelligence Magazine*, 13(4):59–76, 2018.

[21] Hidetoshi Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90(2):227–244, 2000.

[22] Mervyn Stone. Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society: Series B*, 36(2):111–133, 1974.

[23] Masashi Sugiyama, Matthias Krauledat, and Klaus-Robert Müller. Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research*, 8:985–1005, 2007.

[24] Ivan Tomek. Two modifications of CNN. *IEEE Transactions on Systems, Man, and Cybernetics*, 6:769–772, 1976.

[25] Sudhir Varma and Richard Simon. Bias in error estimation when using cross-validation for model selection. *BMC Bioinformatics*, 7(1):1–8, 2006.

[26] Dennis L Wilson. Asymptotic properties of nearest neighbor rules using edited data. *IEEE Transactions on Systems, Man, and Cybernetics*, 2(3):408–421, 1972.

[27] Tzu-Tsung Wong and Po-Yang Yeh. Reliable accuracy estimates from k-fold cross validation. *IEEE Transactions on Knowledge and Data Engineering*, 32(8):1586–1594, 2019.